

## VBScript and Cookies

### What are Cookies?

Web Browsers and Servers use HTTP protocol to communicate and HTTP is a stateless protocol. But for a commercial website, it is required to maintain session information among different pages. For example, one user registration ends after completing many pages. But how to maintain user's session information across all the web pages. In many situations, using cookies is the most efficient method of remembering and tracking preferences, purchases, commissions and other information required for better visitor experience or site statistics.

### How It Works?

Your server sends some data to the visitor's browser in the form of a cookie. The browser may accept the cookie. If it does, it is stored as a plain text record on the visitor's hard drive. Now, when the visitor arrives at another page on your site, the browser sends the same cookie to the server for retrieval. Once retrieved, your server knows/remembers what was stored earlier. Cookies are a plain text data record of 5 variable-length fields –

- **Expires** – The date the cookie will expire. If this is blank, the cookie will expire when the visitor quits the browser.
- **Domain** – The domain name of your site.
- **Path** – The path to the directory or web page that set the cookie. This may be blank if you want to retrieve the cookie from any directory or page.
- **Secure** – If this field contains the word "secure", then the cookie may only be retrieved with a secure server. If this field is blank, no such restriction exists.
- **Name=Value** – Cookies are set and retrieved in the form of key and value pairs.

Cookies were originally designed for CGI programming and cookies' data is automatically transmitted between the web browser and web server, so CGI scripts on the server can read and write cookie values that are stored on the client.

VBScript can also manipulate cookies using the cookie property of the *Document* object. VBScript can read, create, modify and delete the cookie or cookies that apply to the current web page.

### Storing Cookies

The simplest way to create a cookie is to assign a string value to the *document.cookie* object, which looks like this –

### Syntax

```
document.cookie = "key1 = value1;key2 = value2;expires = date"
```

Here *expires* attribute is optional. If you provide this attribute with a valid date or time, then cookie will expire at the given date or time and after that cookies' value will not be accessible.

### Example

Following is the example to set a customer name in *input* cookie.

```
<html>
<head>
  <script type = "text/vbscript">
    Function WriteCookie
      If document.myform.customer.value = "" Then
        msgbox "Enter some value!"
      Else
        cookievalue = (document.myform.customer.value)
        document.cookie = "name = " + cookievalue
        msgbox "Setting Cookies : " & "name = " & cookievalue
      End If
    End Function
  </script>
</head>

<body>
  <form name = "myform" action = "">
    Enter name: <input type = "text" name = "customer"/>
    <input type = "button" value = "Set Cookie" onclick = "WriteCookie()"/>
  </form>
</body>
</html>
```

It will produce the following result. Now enter something in the textbox and press the button "Set Cookie" to set the cookies.

Enter name:

Now, your system has a cookie called *name*. You can set multiple cookies using multiple *key* = *value* pairs separated by comma. You will learn how to read this cookie in next section.

### Reading Cookies

Reading a cookie is just as simple as writing one, because the value of the *document.cookie* object is the cookie. So, you can use this string whenever you want to access the cookie. The *document.cookie* string will keep a list of *name* = *value* pairs separated by semicolons where *name* is the *name* of a cookie and *value* is its string value. You can use strings' *split()* function to break the string into key and values as follows –

### Example

Following is the example to get the cookies set in the previous section –

```
<html>
<head>
  <script type = "text/vbscript">
    Function ReadCookie
      allcookies = document.cookie
```

```

msgbox "All Cookies : " + allcookies
cookiearray = split(allcookies,";")

For i = 0 to ubound(cookiearray)
    Name = Split(cookiearray(i),"=")
    MsgBox "Key is : " + Name(0) + " and Value is : " + Name(1)
Next
End Function
</script>
</head>

<body>
    <form name = "myform" action = "">
        <input type = "button" value = "Get Cookie" onclick = "ReadCookie()" />
    </form>
</body>
</html>

```

**Note** – Here, *UBound* is a method of *Array* class, which returns the length of an array. We will discuss Arrays in a separate chapter; until that time, please try to digest it.

It will produce the following result. Now, press the button "Get Cookie" to see the cookies, which you have set in previous section.

**Note** – There may be some other cookies already set on your machine. So, above code will show you all the cookies set at your machine.

### Setting the Cookies Expiration Date

You can extend the life of a cookie beyond the current browser session by setting an expiration date and saving the expiration date within the cookie. This can be done by setting the *expires* attribute to a date and time.

### Example

The following example illustrates how to set cookie expiration date after 1 Month –

```

<html>
<head>
    <script type = "text/vbscript">
        Function WriteCookie()
            x = now()
            y = dateadd("m",1,now()) ' Making it to expire next
            cookievalue = document.myform.customer.value
            document.cookie = "name = " & cookievalue
            document.cookie = "expires = " & y
            msgbox("Setting Cookies : " & "name=" & cookievalue )
        End Function
    </script>

```

```

</head>
<body>
  <form name = "myform" action = "">
    Enter name: <input type = "text" name = "customer"/>
    <input type = "button" value = "Set Cookie" onclick = "WriteCookie()"/>
  </form>
</body>
</html>

```

## Deleting a Cookie

Sometimes, you will want to delete a cookie so that subsequent attempts to read the cookie return nothing. To do this, you just need to set the expiration date to a time in the past.

## Example

The following example illustrates how to delete a cookie by setting its expiration date 1 Month in the past –

```

<html>
<head>
  <script type = "text/vbscript">
    Function WriteCookie()
      x = now()
      x = now()
      a = Month(x)-1
      b = day(x)
      c = year(x)
      d = DateSerial(c,a,b)
      e = hour(x)

      msgbox e
      f = minute(x)

      msgbox f
      d = cdate(d & " " & e & ":" & f)

      msgbox d
      cookievalue = document.myform.customer.value
      document.cookie = "name = " & cookievalue
      document.cookie = "expires = " & d
      msgbox("Setting Cookies : " & "name=" & cookievalue )
    End Function
  </script>
</head>
<body>
  <form name = "myform" action = "">
    Enter name: <input type = "text" name = "customer"/>
    <input type = "button" value = "Set Cookie" onclick = "WriteCookie()"/>
  </form>

```

```
</body>  
</html>
```